Simultaneous Bayesian inference of motion velocity fields and probabilistic models in successive video-frames described by spatio-temporal MRFs

Yuya Inagaki and Jun-ichi Inoue
Complex Systems Engineering, Graduate School of Information Science and Technology
Hokkaido University, N14-W-9, Kita-ku, Sapporo 060-0814, Japan
Email: inagaki@chaos1.complex.eng.hokudai.ac.jp, j_inoue@complex.eng.hokudai.ac.jp

arXiv:1004.3629v1 [cs.CV] 21 Apr 2010

***Abstract*—We numerically investigate a mean-field Bayesian approach with the assistance of the Markov chain Monte Carlo method to estimate motion velocity fields and probabilistic models simultaneously in consecutive digital images described by spatio-temporal Markov random fields. Preliminary to construction of our procedure, we find that mean-field variables in the iteration diverge due to improper normalization factor of regularization terms appearing in the posterior. To avoid this difficulty, we rescale the regularization term by introducing a scaling factor and optimizing it by means of minimization of the mean-square error. We confirm that the optimal scaling factor stabilizes the mean-field iterative process of the motion velocity estimation. We next attempt to estimate the optimal values of hyper-parameters including the regularization term, which define our probabilistic model macroscopically, by using the Boltzmann-machine type learning algorithm based on gradient descent of marginal likelihood (type-II likelihood) with respect to the hyper-parameters. In our framework, one can estimate both the probabilistic model (hyper-parameters) and motion velocity fields simultaneously. We find that our motion estimation is much better than the result obtained by Zhang and Hanouer (1995) in which the hyper-parameters are set to some ad-hoc values without any theoretical justification.**

I. Introduction

Motion estimation in consecutive video-frames is one of the important techniques in image processing or computer vision community. The motion estimation is defined as estimating the motion velocity fields (vectors) of objects appearing in successive two (video) frames. In the research field of computer vision, the so-called Markov random fields (MRFs for short) have been used to solve the various problems concerning image processing such as image restoration [1], texture analysis and segmentation [2], [3], [4], super-resolution [5], [6] and so on. The MRFs enable us to regularize the ill-posed problems in such a lots of subjects, and then, the original problem can be treated as combinatorial optimization problems under some 'soft' or 'hard' constraints. Actually, Zhang and Hanouer (1995) [7] and Wei and Li (1999) [8] applied the MRFs approach with the assistance of the framework of Bayesian statistics to estimate the motion vector for a given two consecutive digital images. They also utilized the so-called mean-field approximation to carry out the extensive sums in the marginal probability of the posterior and showed that the steady states of the mean-field equations are one of the good candidates for the appropriate motion velocity fields. The same kind of the MRFs approach was implemented by making use of the DSP-based image processing board of SIMD (Single Instruction Multiple Data) machine by Caplier, Luthon and Dumontier (1998) [9] and Luthon, Caplier and Lievin (1999) [10]. They demonstrated that the task to estimate the motion velocity is actually carried out within a realistic time.

In the study by Zhang and Hanouer (1995), they set the so-called hyper-parameters which specify the probabilistic model macroscopically to some ad-hoc values without any reasonable explanation. However, there is no theoretical (statistical) justification for such ad-hoc choices of parameters to estimate the appropriate motion velocity fields. Of course, the selection of hyper-parameters is dependent on a given set of consecutive video-frames and it is important for us to determine the hyper-parameters systematically under some statistical criteria so as to give a fine (if possible, an optimal) average-case performance of the motion estimation.

Taking into account the above requirements from both theoretical and practical sides, from the view point of Bayesian statistics, we examine a mean-field approach with the assistance of the Markov chain Monte Carlo method (the MCMC for short) to estimate both motion velocity fields and hyper-parameters simultaneously in successive video-frames described by spatio-temporal MRFs. We find that mean-field variables in the non-linear maps diverge due to improper normalization factor of regularization terms appearing in the cost function. In order to overcome this difficulty, we rescale the regularization terms by introducing a scaling factor and optimizing it by means of minimization of the mean-square error. We reveal that the optimal scaling factor stabilizes the mean-field iterative procedure of the motion velocity fields estimation. We next attempt to estimate the optimal values of hyper-parameters including the regularization term, which define our probabilistic model macroscopically, by using the *Boltzmann-machine* type learning algorithm based on gradient descent of the marginal likelihood with respect to hyper-parameters. In our framework, one can estimate both the probabilistic model (hyper-parameters) and motion fields simultaneously. We show that our motion estimation is much better than the result given by Zhang and Hanouer (1995) in which hyper-parameters are set to some ad-hoc values without any theoretical explanation.

This paper is organized as follows. In the next section II, we explain our general set-up to deal with the motion velocity estimation by means of spatio-temporal MRFs according to Zhang and Hanouer (1995). From the view point of Bayesian inference, we construct the posterior probability and introduce two kinds of estimations, namely, Maximum A Posteriori (MAP for short) and Maximizer of Posterior Marginal (MPM for short) estimations. In section III, we utilize the mean-field approximation to obtain the MPM estimate and derive the non-linear mean-field equations with respect to the motion velocity fields. As a preliminary, we demonstrate our mean-field approach by setting the hyper-parameters to the values chosen by Zhang and Hanouer (1995) and show that the mean-fields diverge leading up to a quite worse estimation of motion velocity in section IV. To avoid this type of difficulty, we shall rescale the regularization term by introducing a scaling factor and optimizing it by means of minimization of the mean-square error. In section V, we attempt to estimate the optimal values of hyper-parameters including the regularization term, which define our probabilistic model macroscopically, by using the Boltzmann-machine type learning algorithm based on gradient descent of the marginal likelihood with respect to hyper-parameters. In our framework, one can estimate both the probabilistic model (hyper-parameters) and motion velocity fields simultaneously. To proceed to solve the learning equations, we utilize two different ways to carry out the sums coming up exponential order appearing in the learning equations, namely, hybridization of mean-field approximation and MCMC, and simple MCMC. We find that average-case performance of our motion estimation is much better than the result given by Zhang and Hanouer (1995) in which the hyper-parameters are set to some ad-hoc values. The last section is summary.

II. General set-up of motion estimation

In this section, we briefly explain our model system.

A. Spatio-temporal Markov random fields

Let us define a single two-dimensional gray-scale image as a 'video-frame' by $\boldsymbol{x}^{\tau}=\{x_i^{\tau}, i\in S\}$. S denotes a set of pixels in image and index i is related to a point in two-dimensional square lattice (x,y). Here we shall assume that a motion picture consists of successive static images (frames), namely, we distinguish each static image in the motion picture by time index τ as $\boldsymbol{x}^{\tau}$. When we compare the consecutive two static images, that is, $\boldsymbol{x}^{\tau-1}$ and $\boldsymbol{x}^{\tau}$, each pixel in $\boldsymbol{x}^{\tau}$ might change its location with some 'motion velocity'. From this assumption in mind, we introduce velocity fields defined by $\boldsymbol{d}^{\tau}=\{d_i^{\tau}, i\in S\}$. Namely, for each i and for successive two video-frames, a constraint $x_i^{\tau}=x_{i-d_i^{\tau}}^{\tau-1}$ should be satisfied, where 'index' d_i^{τ} is related to a single point $(v_x^{\tau}(i), v_y^{\tau}(i))$ in the two-dimensional vector field. In this paper, we consider that each component of the vector takes a discrete value and the range is limited as $|v_x^{\tau}(i)|, |v_y^{\tau}(i)|\leq d_{\max}-1=5$. It might seem that this range is extremely small in comparison with the range of the grayscales in images (from 0 to 255) or image size ($\sim 30\times 30$), however, if one attempts to construct a detection and alarming system for the dangerous state from 'infinitesimal difference' of patient's breath in ICU (Intensive Care Unit), the limitation of the velocity fields to such a small range is rather desirable (reasonable).

1) Line fields and segmentation fields: Obviously, it is impossible to determine the $\boldsymbol{d}^{\tau}=\{d_i^{\tau}, i\in S\}$ uniquely from just only information about two video-frames $\boldsymbol{x}^{\tau}$ and $\boldsymbol{x}^{\tau-1}$. To compensate this lack information, we introduce line fields and segmentation fields.

The *line fields* guarantee the continuousness between arbitrary two motion velocity fields for the nearest neighboring pixels and we assume that these two motion velocity fields might take similar values. Let us define these line fields by $\boldsymbol{l}=\{l(i,j)|l(i,j)\equiv(h_i,v_i,h_j,v_j)\in S\}$. Here h_i and v_i are labels to represent continuousness between velocity fields in the nearest neighboring (n.n. for short) horizontal and vertical pixels. In other words, we shall define

$$
\begin{aligned}
h_i^{\tau} &= \begin{cases} 0 & (\boldsymbol{d}^{\tau}\text{s for horizontally n.n. pixels are discont.}) \\ 1 & (\boldsymbol{d}^{\tau}\text{s for horizontally n.n. pixels are cont.}) \end{cases} \\
v_i^{\tau} &= \begin{cases} 0 & (\boldsymbol{d}^{\tau}\text{s for vertically n.n. pixels are discont.}) \\ 1 & (\boldsymbol{d}^{\tau}\text{s for vertically n.n. pixels are cont.}) \end{cases}
\end{aligned}
$$

On the other hand, the *segmentation fields* are introduced to distinguish 'predictable areas' and 'unpredictable areas' in the motion velocity fields. Here 'unpredictable areas' means regions hided by some objects before they are moving to somewhere else. Thus, we naturally define the segmentation fields by $\boldsymbol{s}=\{s_i|s_i=0,1\}$ with

$$
s_i^{\tau} = \begin{cases} 0 & (\text{pixel } i \text{ is predictable}) \\ 1 & (\text{pixel } i \text{ is unpredictable}) \end{cases}
$$

B. Bayes rule and posterior probability

In the previous subsections, we defined the motion picture as a series of successive static images by spatio-temporal Markov random fields. To determine the motion velocity fields uniquely, we also introduced the line and segmentation fields. Then, our problem is clearly defined as follows.

Now, our problem is to infer the velocity vector field $\boldsymbol{d}^{\tau}$, line field $\boldsymbol{l}^{\tau}$ and segmentation field $\boldsymbol{s}^{\tau}$ under the condition that two consecutive video-images $\boldsymbol{x}^{\tau}$ and $\boldsymbol{x}^{\tau-1}$ are observed. For the above problem, we easily use the Bayes rule to obtain the posterior probability, which is a probability of $\boldsymbol{\Sigma}^{\tau}\equiv\{\boldsymbol{d}^{\tau},\boldsymbol{s}^{\tau},\boldsymbol{l}^{\tau}\}$ provided that $\boldsymbol{x}^{\tau}$ and $\boldsymbol{x}^{\tau-1}$ are given as

$$
\begin{aligned}
P(\boldsymbol{\Sigma}^{\tau}|\boldsymbol{x}^{\tau},\boldsymbol{x}^{\tau-1}) &= \frac{P(\boldsymbol{x}^{\tau}|\boldsymbol{\Sigma}^{\tau},\boldsymbol{x}^{\tau-1})P(\boldsymbol{\Sigma}^{\tau}|\boldsymbol{x}^{\tau-1})}{\sum_{\boldsymbol{\Sigma}^{\tau}} P(\boldsymbol{x}^{\tau}|\boldsymbol{\Sigma}^{\tau},\boldsymbol{x}^{\tau-1})P(\boldsymbol{\Sigma}^{\tau}|\boldsymbol{x}^{\tau-1})} \\
&= \frac{P(\boldsymbol{x}^{\tau}|\boldsymbol{\Sigma}^{\tau},\boldsymbol{x}^{\tau-1})P(\boldsymbol{\Sigma}^{\tau}|\boldsymbol{x}^{\tau-1})}{P(\boldsymbol{x}^{\tau}|\boldsymbol{x}^{\tau-1})} \qquad (1)
\end{aligned}
$$

where we defined the sums appearing in the above formula by

$\sum_{\boldsymbol{\Sigma}^\tau}(\cdots) \equiv \sum_{\boldsymbol{d}^\tau}(\cdots)\sum_{\boldsymbol{s}^\tau}(\cdots)\sum_{\boldsymbol{l}^\tau}(\cdots)$ with

$$\sum_{\boldsymbol{d}^\tau}(\cdots) \equiv \prod_{i=1}^{N}\sum_{d_i=0}^{d_{\max}-1}(\cdots) \tag{2}$$

$$\sum_{\boldsymbol{s}^\tau}(\cdots) \equiv \prod_{i=1}^{N}\sum_{s_i=0,1}(\cdots) \tag{3}$$

$$\sum_{\boldsymbol{l}^\tau}(\cdots) \equiv \prod_{i=1}^{N}\sum_{h_i=0,1}\sum_{v_i=0,1}(\cdots). \tag{4}$$

For the above posterior, we have the so-called *Maximum A Posteriori (MAP)* estimate by

$$\boldsymbol{\Sigma}_{MAP}^\tau = \arg\max_{\boldsymbol{\Sigma}^\tau}\log P(\boldsymbol{\Sigma}^\tau|\boldsymbol{x}^\tau,\boldsymbol{x}^{\tau-1}) \tag{5}$$

whereas, what we call *Maximizer of Posterior Marginal (MPM)* estimate is given by

$$\Sigma_{i,MPM}^\tau = \arg\max_{\Sigma_i^\tau} P(\Sigma_i^\tau|\boldsymbol{x}^\tau,\boldsymbol{x}^{\tau-1}) = Q(\langle\Sigma_i^\tau\rangle) \tag{6}$$

where we defined the marginal probability by

$$P(\Sigma_i^\tau|\boldsymbol{x}^\tau,\boldsymbol{x}^{\tau-1}) \equiv \sum_{\boldsymbol{\Sigma}^\tau \neq \Sigma_i^\tau} P(\boldsymbol{\Sigma}^\tau|\boldsymbol{x}^\tau,\boldsymbol{x}^{\tau-1}). \tag{7}$$

The average $\langle\cdots\rangle$ appearing in (6) is defined as $\langle\cdots\rangle \equiv \sum_{\boldsymbol{\Sigma}^\tau}(\cdots)P(\boldsymbol{\Sigma}^\tau|\boldsymbol{x}^\tau,\boldsymbol{x}^{\tau-1})$ and $Q(\cdots)$ denotes a function to convert the expectation $\sum_{\boldsymbol{\Sigma}^\tau}\boldsymbol{\Sigma}^\tau P(\boldsymbol{\Sigma}^\tau|\boldsymbol{x}^\tau,\boldsymbol{x}^{\tau-1})$ having a real number into the nearest discrete value.

1) Likelihood function: The likelihood function appearing in the posterior $P(\boldsymbol{x}^\tau|\boldsymbol{\Sigma},\boldsymbol{x}^{\tau-1})$ can be regarded as a probabilistic model to generate the next frame $\boldsymbol{x}^\tau$ provided that the unknown fields $\boldsymbol{\Sigma}$ and the frame in the previous time $\boldsymbol{x}^\tau$ are given. From now on, we omit the τ-dependence of the fields because we consider the motion velocity fields for a given set of just only two consecutive video-frames. Then, we assume $P(\boldsymbol{x}^\tau|\boldsymbol{\Sigma},\boldsymbol{x}^{\tau-1}) \propto \exp\left[-E^{(1)}(\boldsymbol{x}^\tau|\boldsymbol{\Sigma},\boldsymbol{x}^{\tau-1})\right]$ where the cost function $E^{(1)}(\boldsymbol{x}^\tau|\boldsymbol{\Sigma},\boldsymbol{x}^{\tau-1})$ is given by

$$\begin{aligned} E^{(1)}(\boldsymbol{x}^\tau|\boldsymbol{\Sigma},\boldsymbol{x}^{\tau-1}) &= \frac{1}{2\sigma^2}\sum_i(1-s_i)(x_i^\tau - x_{i-d_i}^{\tau-1})^2 \\ &+ \alpha_l \sum_{i,j\in\boldsymbol{N}(i)}\frac{l(i,j)}{(x_i^\tau - x_j^\tau)^2} \end{aligned} \tag{8}$$

where $\boldsymbol{N}(i)$ means a set of nearest neighboring pixels around pixel i. The number of these pixels is $|\boldsymbol{N}(i)| = 4$ (square lattice). The parameters σ and α_l are the so-called *hyper-parameters* which determine the probabilistic model macroscopically.

2) Prior probability: The prior probability $P(\boldsymbol{\Sigma}|\boldsymbol{x}^\tau)$ is a generating model of the fields $\boldsymbol{\Sigma}^\tau$ for a given frame $\boldsymbol{x}^\tau$ and it is given by $P(\boldsymbol{\Sigma}|\boldsymbol{x}^\tau) \propto \exp\left[-E^{(2)}(\boldsymbol{\Sigma}|\boldsymbol{x}^\tau)\right]$ with

$$\begin{aligned} E^{(2)}(\boldsymbol{\Sigma}|\boldsymbol{x}^\tau) &= \lambda_d \sum_{i,j\in\boldsymbol{N}(i)}(1-2\,\mathrm{e}^{-\beta_d\|d_i-d_j\|^2})(1-l(i,j)) \\ &+ \lambda_s \sum_{i,j\in\boldsymbol{N}(i)}(1-l(i,j))(1-2\delta(s_i-s_j)) \\ &+ T_s\sum_i s_i \end{aligned} \tag{9}$$

where we defined the norm $\|\cdots\|$ by

$$\| d_i - d_j \| \equiv \left\{(v_x(i)-v_x(j))^2 + (v_y(i)-v_y(j))^2\right\}^{1/2}$$

and $\lambda_d,\lambda_s,\lambda_l,\beta_d$ and T_s are also hyper-parameters which define the above probabilistic model macroscopically.

3) Posterior: Then, the posterior $P(\boldsymbol{\Sigma}|\boldsymbol{x}^\tau,\boldsymbol{x}^{\tau-1})$, namely, the probability of the desired fields for a given set of two successive video-frames $\boldsymbol{x}^\tau,\boldsymbol{x}^{\tau-1}$ is constructed by the product of likelihood $P(\boldsymbol{x}^\tau|\boldsymbol{\Sigma},\boldsymbol{x}^{\tau-1})$ and prior $P(\boldsymbol{\Sigma}|\boldsymbol{x}^{\tau-1})$, that is $P(\boldsymbol{\Sigma}|\boldsymbol{x}^\tau,\boldsymbol{x}^{\tau-1}) \propto P(\boldsymbol{x}^\tau|\boldsymbol{\Sigma},\boldsymbol{x}^{\tau-1})P(\boldsymbol{\Sigma}|\boldsymbol{x}^{\tau-1})$.

By means of the cost function, we have

$$\begin{aligned} P(\boldsymbol{\Sigma}|\boldsymbol{x}^\tau,\boldsymbol{x}^{\tau-1}) &\propto \exp\left[-E^{(1)}(\boldsymbol{x}^\tau|\boldsymbol{\Sigma},\boldsymbol{x}^{\tau-1}) - E^{(2)}(\boldsymbol{\Sigma}|\boldsymbol{x}^\tau)\right] \\ &\equiv \exp\left[-E(\boldsymbol{\Sigma}|\boldsymbol{x}^\tau,\boldsymbol{x}^{\tau-1})\right]. \end{aligned} \tag{10}$$

The total cost of the system, which is now defined by $-\log P(\boldsymbol{\Sigma}|\boldsymbol{x}^\tau,\boldsymbol{x}^{\tau-1})$, is written as

$$\begin{aligned} E(\boldsymbol{\Sigma}|\boldsymbol{x}^\tau,\boldsymbol{x}^{\tau-1}) &\equiv \frac{1}{2\sigma^2}\sum_i(1-s_i)(x_i^\tau - x_{i-d_i}^{\tau-1})^2 \\ &+ \lambda_d \sum_{i,j\in\boldsymbol{N}(i)}(1-2\,\mathrm{e}^{-\beta_d\|d_i-d_j\|^2})(1-l(i,j)) \\ &+ \lambda_s \sum_{i,j\in\boldsymbol{N}(i)}(1-l(i,j))(1-2\delta(s_i-s_j)) \\ &+ \alpha_l \sum_{i,j\in\boldsymbol{N}(i)}\frac{l(i,j)}{(x_i^\tau - x_j^\tau)^2} + T_s\sum_i s_i \end{aligned} \tag{11}$$

where the first term appearing in the right hand side of the above cost function is introduced to prevent pixel $x_i^{\tau-1}$ at the location i from moving to the position $i-d_i^\tau$ where is quite far from i. The second term confirms the continuousness between velocity vectors for the nearest neighboring pixels and we easily find that the term is identical to the Hamiltonian (energy function) for the so-called dynamically diluted *ferromagnetic Q-Ising model* in the literature of statistical physics, that is to say, we have

$$\begin{aligned} &\lambda_d \sum_{i,j\in\boldsymbol{N}(i)}(1-l(i,j))(1-2\,\mathrm{e}^{-\beta_d\|d_i-d_j\|^2}) \\ &\simeq 2\lambda_d\beta_d \sum_{i,j\in\boldsymbol{N}(i)}(1-l(i,j))\,\| d_i - d_j \|^2 \\ &+ \boldsymbol{d}\text{-independent const.} \end{aligned} \tag{12}$$

in the limit of $\beta_d \to 0$. The third term in (11) denotes a correlation between the line and the segmentation fields. The forth term represents a correlation between the line fields

and the distance of pixels located in the nearest neighboring positions. The last term controls the number of non-zero segmentation fields and this term can be regarded as the so-called *chemical potential* in the literature of statistical physics.

III. Mean-field equations on pixel

In the previous section, we constructed the posterior by making use of the Bayes rule. Therefore, we can use both MAP and MPM estimations by means of (5) and (6), respectively. Here we should notice that the MAP estimate is recovered by means of

$$\Sigma_{i,MAP} = \lim_{\beta\to\infty} Q(\langle\Sigma_i\rangle_\beta),\ \langle\cdots\rangle_\beta \equiv \sum_{\mathbf{\Sigma}}(\cdots)P_\beta(\mathbf{\Sigma}|\boldsymbol{x}^\tau,\boldsymbol{x}^{\tau-1})$$

with $P_\beta(\mathbf{\Sigma}|\boldsymbol{x}^\tau,\boldsymbol{x}^{\tau-1}) \propto \exp\left[-\beta E(\mathbf{\Sigma}|\boldsymbol{x}^\tau,\boldsymbol{x}^{\tau-1})\right]$. From the above definitions, the MPM estimate is obtained by $\Sigma_{i,MPM} = Q(\langle\Sigma_i\rangle_1)$. Therefore, our problem now seems to be completely solved. However, the number of sums appearing in the expectation $\langle\cdots\rangle_\beta$

$$\begin{aligned}\sum_{\mathbf{\Sigma}}(\cdots) &= \sum_{s_1=0,1}\cdots\sum_{s_N=0,1}\sum_{d_1=0}^{d_{\max}-1}\cdots\sum_{d_N=0}^{d_{\max}-1} \\ &\times \sum_{l_1=0,1}\cdots\sum_{l_N=0,1}(\cdots)\end{aligned} \quad (13)$$

comes up to exponential order as $\mathrm{e}^{N\log 4d_{\max}}$. Obviously, it is impossible for us to carry out the sums even for the system size is $N = 30\times 30 = 900$ within a realistic time.

Then, we use the mean-field approximation to overcome this type of computational difficulties. Namely, we rewrite the cost function by replacing the motion velocity fields with the corresponding expectations except for a single component of the fields. For instance, for say s_i, we have the mean-field approximated cost function as follows.

$$\begin{aligned}E &\simeq E^0(s_i) \equiv -\frac{s_i}{2\sigma^2}(x_i^\tau - x_{i-\langle d_i\rangle_\beta^{\rm mf}}^{\tau-1})^2 + T_s s_i \\ &+ \lambda_s\sum_{j\in\boldsymbol{N}(i)}(1-\langle l(i,j)\rangle_\beta^{\rm mf})(1-\delta(s_i-\langle s_j\rangle_\beta^{\rm mf}))\end{aligned}$$

By using the same way as s_i, we have for d_i as

$$\begin{aligned}E &\simeq E^0(d_i) \equiv \frac{(1-\langle s_i\rangle_\beta^{\rm mf})}{2\sigma^2}(x_i^\tau - x_{i-d_i}^{\tau-1})^2 \\ &+ \lambda_d\sum_{j\in\boldsymbol{N}(i)}(1-2\,\mathrm{e}^{-\beta_d\|d_i-\langle d_j\rangle_\beta^{\rm mf}\|^2})(1-\langle l(i,j)\rangle_\beta^{\rm mf})\end{aligned}$$

and obtain for $l(i,j)$ as

$$\begin{aligned}&E \simeq E^0(l(i,j)) \\ &\equiv \lambda_d(1-2\,\mathrm{e}^{-\beta_d\|\langle d_i\rangle_\beta^{\rm mf}-\langle d_j\rangle_\beta^{\rm mf}\|^2})(1-l(i,j)) \\ &+ \lambda_s(1-l(i,j))(1-2\delta(\langle s_i\rangle_\beta^{\rm mf}-\langle s_j\rangle_\beta^{\rm mf})) + \frac{\alpha_l\, l(i,j)}{(x_i^\tau - x_j^\tau)^2}\end{aligned}$$

where $\delta(\cdots)$ stands for a delta-function. By means of the above approximated cost functions, one obtains the following self-consistent equations for $\forall_{i,j\in S}$.

$$\begin{aligned}\langle s_i\rangle_\beta^{\rm mf} &= \frac{\sum_{s_i=0}^1 s_i\,\mathrm{e}^{-\beta E^0(s_i)}}{\sum_{s_i=0}^1 \mathrm{e}^{-\beta E^0(s_i)}} \\ &\equiv \Phi_\beta^s(\langle d_i\rangle_\beta^{\rm mf},\langle l(i,j)\rangle_\beta^{\rm mf},\langle s_i\rangle_\beta^{\rm mf},\cdots) \\ \langle d_i\rangle_\beta^{\rm mf} &= \frac{\sum_{d_i=0}^{d_{\max}-1} d_i\,\mathrm{e}^{-\beta E^0(d_i)}}{\sum_{d_i=0}^{d_{\max}-1}\mathrm{e}^{-\beta E^0(d_i)}} \\ &\equiv \Phi_\beta^d(\langle s_i\rangle_\beta^{\rm mf},\langle d_j\rangle_\beta^{\rm mf},\langle l(i,j)\rangle_\beta^{\rm mf},\cdots) \\ \langle l(i,j)\rangle_\beta^{\rm mf} &= \frac{\sum_{l(i,j)=0}^1 l(i,j)\,\mathrm{e}^{-\beta E^0(l(i,j))}}{\sum_{l(i,j)=0}^1\mathrm{e}^{-\beta E^0(l(i,j))}} \\ &\equiv \Phi_\beta^l(\langle d_i\rangle_\beta^{\rm mf},\langle d_j\rangle_\beta^{\rm mf},\langle s_i\rangle_\beta^{\rm mf},\langle s_j\rangle_\beta^{\rm mf},\cdots)\end{aligned}$$

Regarding the above self-consistent equations with respect to single-site averages as the following 'non-linear maps':

$$\langle s_i\rangle_\beta^{{\rm mf}(t+1)} = \Phi_\beta^s(\langle d_i\rangle_\beta^{{\rm mf}(t)},\langle l(i,j)\rangle_\beta^{{\rm mf}(t)},\langle s_i\rangle_\beta^{{\rm mf}(t)},\cdots) \quad (14)$$

$$\langle d_i\rangle_\beta^{{\rm mf}(t+1)} = \Phi_\beta^d(\langle s_i\rangle_\beta^{{\rm mf}(t)},\langle d_j\rangle_\beta^{{\rm mf}(t)},\langle l(i,j)\rangle_\beta^{{\rm mf}(t)},\cdots) \quad (15)$$

$$\begin{aligned}&\langle l(i,j)\rangle_\beta^{{\rm mf}(t+1)} \\ &= \Phi_\beta^l(\langle d_i\rangle_\beta^{{\rm mf}(t)},\langle d_j\rangle_\beta^{{\rm mf}(t)},\langle s_i\rangle_\beta^{{\rm mf}(t)},\langle s_j\rangle_\beta^{{\rm mf}(t)},\cdots)\end{aligned} \quad (16)$$

we look for the steady states of the above maps which should satisfy the following convergence condition.

$$\begin{aligned}\epsilon_t &\equiv N^{-1}\{\|\,\langle\boldsymbol{s}\rangle_\beta^{{\rm mf}(t)}-\langle\boldsymbol{s}\rangle_\beta^{{\rm mf}(t-1)}\,\|^2 \\ &+ \|\,\langle\boldsymbol{d}\rangle_\beta^{{\rm mf}(t)}-\langle\boldsymbol{d}\rangle_\beta^{{\rm mf}(t-1)}\,\|^2 \\ &+ \|\,\langle\boldsymbol{l}\rangle_\beta^{{\rm mf}(t)}-\langle\boldsymbol{l}\rangle_\beta^{{\rm mf}(t-1)}\,\|^2\}^{1/2} < \epsilon\end{aligned} \quad (17)$$

where ϵ should be a small value, say $\epsilon = 1.0\times 10^{-5}$. In general, a control parameter β is time-dependent variable as $\beta(t)$ and the MAP estimate is obtained by controlling it as $\beta(t)\to\infty$ as $t\to\infty$. On the other hand, the MPM estimate is constructed by setting the β to 1 during the above iterations.

Generally speaking, the steady state $\langle\cdots\rangle_\beta^{{\rm mf}(\infty)}$ is different from $\langle\cdots\rangle_\beta$ which is a solution of the self-consistent equations, however, it might assume that the $\langle\cdots\rangle_\beta^{{\rm mf}(\infty)}$ more likely to be close to $\langle\cdots\rangle_\beta$ if the landscape of the cost is not so complicated like *spin glasses* [11].

IV. Preliminary : divergence of mean-fields

To check the usefulness of the above procedure, we examine our mean-field algorithm to infer the motion velocity fields for a given set of two successive frames shown in Fig. 1. It should be noted that these two frames are artificially given and obviously, the true motion velocity vector fields are now explicitly provided for us to check the usefulness of our mean-field algorithm.

Generally speaking in the Bayesian inference, setting the hyper-parameters appearing in the probabilistic model is one of the quite important tasks and here we examine the values $(\beta,\sigma^2,\lambda_d,\beta_d,\alpha_l,T_s,\lambda_s) = (1,0.2,2.5,4,200,5,2)$ which were given ad-hoc by Zhang and Hanouer (1995). We find

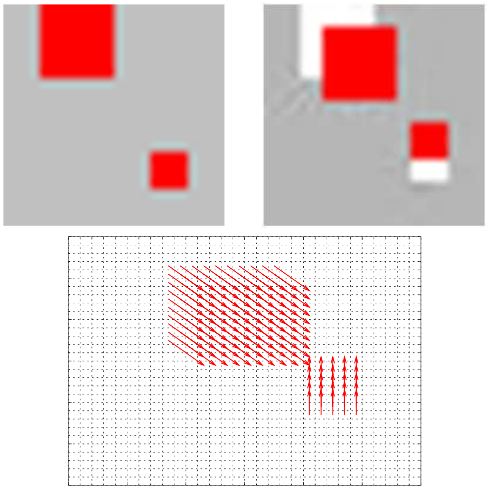

Fig. 1. Typical artificial images as a set of successive two video-frames. Image before moving (upper left) and image after moving (upper right). The lower panel shows 'true' motion velocity fields for the situation given by the upper panels. In the above images, arbitrary grayscales are given to the segmentation areas and the region in which the objects are located.

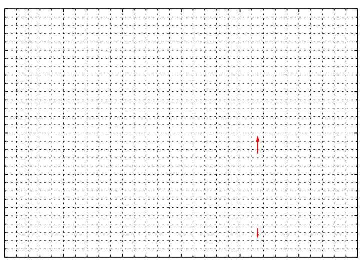

Fig. 2. The resultant velocity fields calculated by the choice of hyper-parameters $(\beta, \sigma^2, \lambda_d, \beta_d, \alpha_l, T_s, \lambda_s) = (1, 0.2, 2.5, 4, 200, 5, 2)$. The velocity fields shrink to a few points with small lengths.

that for the above choice of the hyper-parameter causes a divergence of the mean-fields such as $\langle s_i \rangle_\beta^{\rm mf}$ due to the regularization terms $(1/2\sigma^2)(1 - \langle s_i \rangle_\beta^{\rm mf})(x_i^\tau - x_{i-d_i}^{\tau-1})^2$ or $-(s_i/2\sigma^2)(x_i^\tau - x_{i-\langle s_i \rangle_\beta^{\rm mf}}^{\tau-1})^2$ which appear in the mean-field equations. We show the resultant velocity fields calculated by the above choice of hyper-parameters in Fig. 2. We find that the velocity fields shrink to a few points with small lengths and one apparently fails to estimate the true velocity fields.

A. Optimization of scaling factor

The origin of the above difficulty apparently comes from the divergence of these regularization terms evaluated for two extremely different values of pixels, for instance, say $x_i^\tau = 255$ and $x_{i-d_i^\tau}^{\tau-1} = 0$ which leads to $\mathrm{e}^{(255-0)^2} \sim \infty$. This fact tells us that there exist several serious cases (combinations of two consecutive video-frames) for which the ad-hoc hyper-parameter selection causes this type of divergence during the iteration of mean-field equations.

To avoid the essential difficulty, we rescale the hyper-parameter σ^2 as $\sigma^2 \mapsto \mu\sigma^2$ and optimizing the scaling factor μ from the view point of several different performance measures.

1) Performance measures: We first introduce two different kinds of mean-square errors as average-case performance measures to determine the optimal scaling factor μ.

$$D_1(\mu) \equiv \frac{1}{N_1}\sum_{i=1}^{N}(1-s_i) \parallel d_i^{(0)} - d_i \parallel^2 \qquad (18)$$

$$D_2(\mu) \equiv \frac{1}{N_2}\sum_{i=1}^{N} s_i \parallel d_i^{(0)} - d_i \parallel^2 \qquad (19)$$

where $N_1 \equiv \sum_{i=1}^{N}(1-s_i)$, $N_2 \equiv \sum_{i=1}^{N} s_i$ and we should keep in mind that $N = N_1 + N_2$ holds. $\boldsymbol{d}^{(0)}$ is a true velocity field for a given set of two successive images shown in Fig. 1. Thus, the D_1 denotes the mean-square error defined by the difference between the true and the estimated velocity fields for zero segmentation regions. On the other hand, D_2 is the mean-square error evaluated for non-zero segmentation regions.

We also introduce the bit-error rate which is defined as the number of estimated pixels which are different from the true ones. Namely, we use

$$\delta_1(\mu) \equiv \frac{1}{N_1}\sum_{i=1}^{N}(1-s_i)\,\hat{\delta}_{d_i^0, d_i} \qquad (20)$$

$$\delta_2(\mu) \equiv \frac{1}{N_2}\sum_{i=1}^{N} s_i\,\hat{\delta}_{d_i^0, d_i} \qquad (21)$$

where $\hat{\delta}_{x,y}$ means a Kronecker's delta which is defined by

$$\hat{\delta}_{d_i^0, d_i} \equiv \delta_{v_x^0(i), v_x(i)}\delta_{v_y^0(i), v_y(i)} \qquad (22)$$

where $\delta_{x,y}$ is a 'conventional' Kronecker's delta. In Fig. 3,

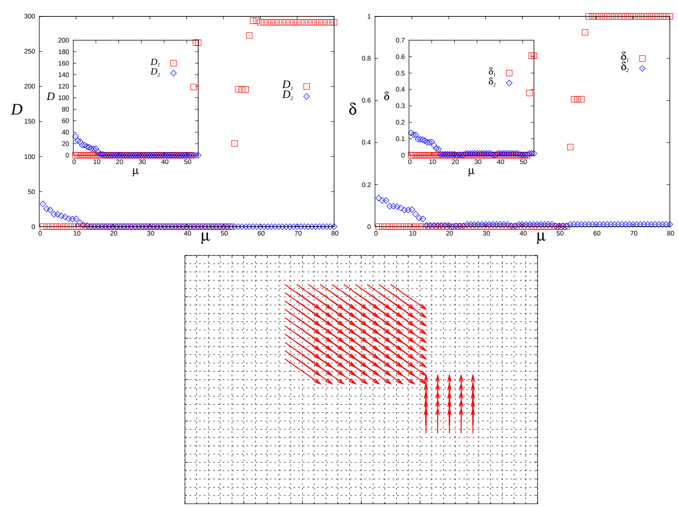

Fig. 3. Behaviour of two kinds of the mean-square errors D_1, D_2 (upper left), the bit-error rates δ_1, δ_2 (upper right) as a function of scaling factor μ. The lower panel shows the resultant velocity fields obtained by setting the optimal scaling factor $\mu_* \simeq 21$. The grayscale levels of the background and segmentation areas are $Q = 0$ and $Q = 40$, respectively. The grayscale levels for the moving object are distributed within the range $Q = 10 \sim 30$.

we plot the behaviour of two kinds of the mean-square errors

D_1, D_2 (upper left), the bit-error rates δ_1, δ_2 (upper right) as a function of scaling factor μ. The lower panel shows the resultant velocity fields obtained by setting the optimal scaling factor $\mu_* \simeq 21$. From these panels, we find that the resultant velocity fields are very close to the true fields when we set the scaling factor appropriately. However, the ad-hoc choice of the other hyper-parameters $(\beta, \sigma^2, \lambda_d, \beta_d, \alpha_l, T_s, \lambda_s)$ should not be confirmed for the best possible velocity fields estimation for a given other set of the successive images. To make matter worse, in practice, we can use neither mean-square error nor bit-error rate because these quantities require the information about the true fields $\boldsymbol{d}^{(0)}$ (for instance, see the definition of D_1). Therefore, we should seek some theoretical justifications to determine the optimal hyper-parameters.

V. Maximum marginal likelihood criteria

In statistics, in order to determine the hyper-parameters $\boldsymbol{\Xi} \equiv \{\mu, \sigma, \lambda_d, \lambda_s, T_s, \alpha_l, \beta_d\}$ of the probabilistic model which contains latent variables $\boldsymbol{\Sigma} \equiv \{\boldsymbol{s}, \boldsymbol{d}, \boldsymbol{l}\}$, the so-called *maximum marginal likelihood estimation* is widely used. The marginal likelihood (the type-II likelihood) is defined by

$$-F_{\boldsymbol{\Xi}}(\boldsymbol{x}^\tau, \boldsymbol{x}^{\tau-1}) \equiv \log \sum_{\boldsymbol{\Sigma}} P(\boldsymbol{\Sigma}|\boldsymbol{x}^\tau, \boldsymbol{x}^{\tau-1}) \qquad (23)$$

namely, the marginal likelihood is obtained by taking the sums of these latent variables in the (log) likelihood function. It should be noted that the above marginal likelihood is dependent on the 'input' two successive frames $\boldsymbol{x}^\tau, \boldsymbol{x}^{\tau-1}$. We can easily show that the marginal likelihood is maximized at the true values of the hyper-parameters $\boldsymbol{\Xi}^0$, namely,

$$\left[-F_{\boldsymbol{\Xi}^0}(\boldsymbol{x}^\tau, \boldsymbol{x}^{\tau-1})\right]_{\boldsymbol{x}^\tau, \boldsymbol{x}^{\tau-1}} \geq \left[-F_{\boldsymbol{\Xi}}(\boldsymbol{x}^\tau, \boldsymbol{x}^{\tau-1})\right]_{\boldsymbol{x}^\tau, \boldsymbol{x}^{\tau-1}}. \qquad (24)$$

where we defined the observable data-average by $[\cdots] \equiv \sum_{\boldsymbol{x}^\tau, \boldsymbol{x}^{\tau-1}} (\cdots) P_{\boldsymbol{\Xi}_0}(\boldsymbol{x}^\tau, \boldsymbol{x}^{\tau-1})$.

A. Kullback-Leibler information

Taking into account the fact that the Kullback-Leibler (KL) information can not be negative, we can easily show the inequality (24).

Let us consider the KL information between the true probabilistic model $P_{\boldsymbol{\Xi}_0}(\boldsymbol{x}^\tau, \boldsymbol{x}^{\tau-1})$ and the model $P_{\boldsymbol{\Xi}}(\boldsymbol{x}^\tau, \boldsymbol{x}^{\tau-1})$. Then, from the definition of the KL information, we immediately have

$$\begin{aligned}
& KL(P_{\boldsymbol{\Xi}_0}||P_{\boldsymbol{\Xi}}) \\
= & \sum_{\boldsymbol{x}^\tau, \boldsymbol{x}^{\tau-1}} P_{\boldsymbol{\Xi}_0}(\boldsymbol{x}^\tau, \boldsymbol{x}^{\tau-1}) \log \left\{ \frac{P_{\boldsymbol{\Xi}_0}(\boldsymbol{x}^\tau, \boldsymbol{x}^{\tau-1})}{P_{\boldsymbol{\Xi}}(\boldsymbol{x}^\tau, \boldsymbol{x}^{\tau-1})} \right\} \\
= & \sum_{\boldsymbol{x}^\tau, \boldsymbol{x}^{\tau-1}} P_{\boldsymbol{\Xi}_0}(\boldsymbol{x}^\tau, \boldsymbol{x}^{\tau-1}) \log P_{\boldsymbol{\Xi}_0}(\boldsymbol{x}^\tau, \boldsymbol{x}^{\tau-1}) \\
- & \sum_{\boldsymbol{x}^\tau, \boldsymbol{x}^{\tau-1}} P_{\boldsymbol{\Xi}_0}(\boldsymbol{x}^\tau, \boldsymbol{x}^{\tau-1}) \log P_{\boldsymbol{\Xi}}(\boldsymbol{x}^\tau, \boldsymbol{x}^{\tau-1}) \\
= & [-F_{\boldsymbol{\Xi}_0}(\boldsymbol{x}^\tau, \boldsymbol{x}^{\tau-1})]_{\boldsymbol{x}^\tau, \boldsymbol{x}^{\tau-1}} - [-F_{\boldsymbol{\Xi}}(\boldsymbol{x}^\tau, \boldsymbol{x}^{\tau-1})]_{\boldsymbol{x}^\tau, \boldsymbol{x}^{\tau-1}} \\
\geq & \; 0. \qquad (25)
\end{aligned}$$

The equality holds if and only if $\boldsymbol{\Xi} = \boldsymbol{\Xi}_0$. Therefore, the inequality (24) holds and this means that the marginal likelihood takes its maximum at the true values of the hyper-parameters. We use this fact to determine the hyper-parameters. In other words, the marginal likelihood is regarded as a 'cost function' whose lowest energy states might be a candidate of the true hyper-parameters.

VI. Hyper-parameter estimation

As we saw in the previous section, we should determine hyper-parameters so as to minimize the marginal likelihood. In this section, we attempt to construct the Boltzmann-machine type learning equations which are derived by means of taking a gradient of the marginal likelihood with respect to the hyper-parameters.

A. Boltzmann-machine learning and its dynamics

Let us define $\boldsymbol{C}(\boldsymbol{\Sigma})$ as a conjugate statistics for the parameter $\boldsymbol{\Xi}$. Then, the Boltzmann-machine learning equation is obtained as

$$\begin{aligned}
\frac{d\boldsymbol{\Xi}}{dt} &= -\frac{\partial F_{\boldsymbol{\Xi}}(\boldsymbol{x}^\tau, \boldsymbol{x}^{\tau-1})}{\partial \boldsymbol{\Xi}} \\
&= -\frac{\sum_{\boldsymbol{\Sigma}} \boldsymbol{C}(\boldsymbol{\Sigma}) P(\boldsymbol{\Sigma}|\boldsymbol{x}^\tau, \boldsymbol{x}^{\tau+1})}{\sum_{\boldsymbol{\Sigma}} P(\boldsymbol{\Sigma}|\boldsymbol{x}^\tau, \boldsymbol{x}^{\tau+1})} \qquad (26)
\end{aligned}$$

Namely, we have

$$\frac{dB}{dt} = -\frac{\sum_{\boldsymbol{\Sigma}} \{\sum_i (1-s_i)(x_i^\tau - x_{i-d_i}^{\tau-1})^2\} \mathrm{e}^{-U}}{\sum_{\boldsymbol{\Sigma}} \mathrm{e}^{-U}} \qquad (27)$$

$$\frac{d\lambda_d}{dt} = -\frac{\sum_{\boldsymbol{\Sigma}} \Lambda_d^{\beta_d}(d_i, d_j, l(i,j)) \mathrm{e}^{-U}}{\sum_{\boldsymbol{\Sigma}} \mathrm{e}^{-U}} \qquad (28)$$

$$\frac{d\lambda_s}{dt} = -\frac{\sum_{\boldsymbol{\Sigma}} \Lambda_d^{\beta_d}(d_i, d_j, l(i,j)) \mathrm{e}^{-U}}{\sum_{\boldsymbol{\Sigma}} \mathrm{e}^{-U}} \qquad (29)$$

$$\frac{d\alpha_l}{dt} = -\frac{\sum_{\boldsymbol{\Sigma}} \left\{\sum_{i,j \in \boldsymbol{N}(i)} \frac{l(i,j)}{(x_i^\tau - x_j^\tau)^2}\right\} \mathrm{e}^{-U}}{\sum_{\boldsymbol{\Sigma}} \mathrm{e}^{-U}} \qquad (30)$$

$$\frac{d\beta_d}{dt} = -\frac{\sum_{\boldsymbol{\Sigma}} \mathcal{B}_d^{\beta_d}(l(i,j), d_i, d_j) \mathrm{e}^{-U}}{\sum_{\boldsymbol{\Sigma}} \mathrm{e}^{-U}} \qquad (31)$$

$$\frac{dT_l}{dt} = -\frac{\sum_{\boldsymbol{\Sigma}} \{\sum_i s_i\} \mathrm{e}^{-U}}{\sum_{\boldsymbol{\Sigma}} \mathrm{e}^{-U}} \qquad (32)$$

where we defined

$$\Lambda_d^{\beta_d}(d_i, d_j, l(i,j)) \equiv \sum_{i,j \in \boldsymbol{N}(i)} (1 - 2\,\mathrm{e}^{-\beta_d \|d_i - d_j\|^2})(1 - l(i,j)) \qquad (33)$$

$$\mathcal{B}_d^{\beta_d}(l(i,j), d_i, d_j) \equiv \sum_{i,j \in \boldsymbol{N}(i)} (1 - l(i,j)) \parallel d_i - d_j \parallel^2 \mathrm{e}^{-\beta_d \|d_i - d_j\|} \qquad (34)$$

$B \equiv 1/2\mu\sigma^2$ and $U \equiv \beta E(\boldsymbol{\Sigma}|\boldsymbol{x}^\tau, \boldsymbol{x}^{\tau-1})$. It should be noticed that the number of sums appearing in the right hand sides of the above equations comes up to exponential order and it is impossible for us to carry out them.

B. Hybridization of mean-field approximation and MCMC

To overcome this computational difficulty, we utilize the mean-field approximation. We first replace the variables $\mathbf{\Sigma}$ with the corresponding expectations expect for the variables appearing in the brackets $\{\cdots\}$ in the right hand side of the learning equations. For instance, $dB/dt = -\partial F_{\mathbf{\Xi}}/\partial B$ now leads to

$$\frac{dB}{dt} = -\frac{\sum_{s_i,d_i}\left\{\sum_i(1-s_i)(x_i^{\tau}-x_{i-d_i}^{\tau-1})^2\right\}\mathrm{e}^{-\langle U\rangle^{\mathrm{mf}}_{s_i,d_i}}}{\sum_{s_i,d_i}\mathrm{e}^{-\langle U\rangle^{\mathrm{mf}}_{s_i,d_i}}} \tag{35}$$

$$\begin{aligned}\langle U\rangle^{\mathrm{mf}}_{s_i,d_i} \equiv{}& B\sum_i(1-s_i)(x_i^{\tau}-x_{i-d_i}^{\tau-1})^2\\ &+ \lambda_d\sum_{i,j\in\boldsymbol{N}(i)}(1-2\,\mathrm{e}^{-\beta_d\|d_i-\langle d_j\rangle^{\mathrm{mf}}\|^2})(1-\langle l(i,j)\rangle^{\mathrm{mf}})\\ &+ \lambda_s\sum_{i,j\in\boldsymbol{N}(i)}(1-\langle l(i,j)\rangle^{\mathrm{mf}})(1-2\delta(s_i-\langle s_j\rangle^{\mathrm{mf}}))\\ &+ \alpha_l\sum_{i,j\in\boldsymbol{N}(i)}\frac{\langle l(i,j)\rangle^{\mathrm{mf}}}{(x_i^{\tau}-x_j^{\tau})^2}+T_s\sum_i s_i\end{aligned} \tag{36}$$

where we set $\beta=1$, namely, we calculate the MPM estimate in our framework. Using the same way as the above, $d\lambda_d/dt = -\partial F_{\mathbf{\Xi}}/\partial\lambda_d$ leads to

$$\frac{d\lambda_d}{dt} = -\frac{\sum_{d_i,d_j,l_{ij}}\Lambda_d^{\beta_d}(d_i,d_j,l(i,j))\mathrm{e}^{-\langle U\rangle^{\mathrm{mf}}_{d_i,d_j,l_{ij}}}}{\sum_{d_i,d_j,l_{ij}}\mathrm{e}^{-\langle U\rangle^{\mathrm{mf}}_{d_i,d_j,l_{ij}}}} \tag{37}$$

$$\begin{aligned}&\Lambda_d^{\beta_d}(d_i,d_j,l(i,j))\\ &\quad\equiv \sum_{i,j\in\boldsymbol{N}(i)}(1-2\,\mathrm{e}^{-\beta_d\|d_i-d_j\|^2})(1-l(i,j))\end{aligned} \tag{38}$$

$$\begin{aligned}\langle U\rangle^{\mathrm{mf}}_{d_i,d_j,l_{ij}} \equiv{}& B\sum_i(1-\langle s_i\rangle^{\mathrm{mf}})(x_i^{\tau}-x_{i-d_i}^{\tau-1})^2\\ &+ \lambda_d\sum_{i,j\in\boldsymbol{N}(i)}(1-2\,\mathrm{e}^{-\beta_d\|d_i-d_j\|^2})(1-l(i,j))\\ &+ \lambda_s\sum_{i,j\in\boldsymbol{N}(i)}(1-l(i,j))(1-2\delta(\langle s_i\rangle^{\mathrm{mf}}-\langle s_j\rangle^{\mathrm{mf}}))\\ &+ \alpha_l\sum_{i,j\in\boldsymbol{N}(i)}\frac{l(i,j)}{(x_i^{\tau}-x_j^{\tau})^2}+T_s\sum_i\langle s_i\rangle^{\mathrm{mf}}\end{aligned} \tag{39}$$

The equations for the other parameters are also rewritten as

$$\frac{d\lambda_s}{dt} = -\frac{\sum_{l_{ij},s_i,s_j}\Lambda_s(l(i,j),s_i,s_j)\mathrm{e}^{-\langle U\rangle^{\mathrm{mf}}_{l_{ij},s_i,s_j}}}{\sum_{l_{ij},s_i,s_j}\mathrm{e}^{-\langle U\rangle^{\mathrm{mf}}_{l_{ij},s_i,s_j}}} \tag{40}$$

$$\begin{aligned}&\Lambda_s(l(i,j),s_i,s_j)\\ &\quad\equiv \sum_{i,j\in\boldsymbol{N}(i)}(1-l(i,j))(1-2\delta(s_i-s_j))\end{aligned} \tag{41}$$

$$\begin{aligned}\langle U\rangle^{\mathrm{mf}}_{s_i,s_j,l_{ij}} \equiv{}& B\sum_i(1-s_i)(x_i^{\tau}-x_{i-\langle d_i\rangle^{\mathrm{mf}}}^{\tau-1})^2\\ &+ \lambda_d\sum_{i,j\in\boldsymbol{N}(i)}(1-2\,\mathrm{e}^{-\beta_d\|\langle d_i\rangle^{\mathrm{mf}}-\langle d_j\rangle^{\mathrm{mf}}\|^2})(1-l(i,j))\\ &+ \lambda_s\sum_{i,j\in\boldsymbol{N}(i)}(1-l(i,j))(1-2\delta(s_i-s_j))\\ &+ \alpha_l\sum_{i,j\in\boldsymbol{N}(i)}\frac{l(i,j)}{(x_i^{\tau}-x_j^{\tau})^2}+T_s\sum_i s_i\end{aligned} \tag{42}$$

$$\frac{d\alpha_l}{dt} = -\frac{\sum_{l_{ij}}\left\{\sum_{i,j\in\boldsymbol{N}(i)}\frac{l(i,j)}{(x_i^{\tau}-x_j^{\tau})^2}\right\}\mathrm{e}^{-\langle U\rangle^{\mathrm{mf}}_{l_{ij}}}}{\sum_{l_{ij}}\mathrm{e}^{-\langle U\rangle^{\mathrm{mf}}_{l_{ij}}}} \tag{43}$$

$$\begin{aligned}\langle U\rangle^{\mathrm{mf}}_{l_{ij}} \equiv{}& B\sum_i(1-\langle s_i\rangle)(x_i^{\tau}-x_{i-\langle d_i\rangle}^{\tau-1})^2\\ &+ \lambda_d\sum_{i,j\in\boldsymbol{N}(i)}(1-2\,\mathrm{e}^{-\beta_d\|\langle d_i\rangle^{\mathrm{mf}}-\langle d_j\rangle^{\mathrm{mf}}\|^2})\\ &\times (1-l(i,j))\\ &+ \lambda_s\sum_{i,j\in\boldsymbol{N}(i)}(1-l(i,j))\\ &\times (1-2\delta(\langle s_i\rangle^{\mathrm{mf}}-\langle s_j\rangle^{\mathrm{mf}}))\\ &+ \alpha_l\sum_{i,j\in\boldsymbol{N}(i)}\frac{l(i,j)}{(x_i^{\tau}-x_j^{\tau})^2}+T_s\sum_i\langle s_i\rangle^{\mathrm{mf}}\end{aligned} \tag{44}$$

$$\frac{d\beta_d}{dt} = -\frac{\sum_{d_i,d_j,l_{ij}}\mathcal{B}_d^{\beta_d}(l(i,j),d_i,d_j)\mathrm{e}^{-\langle U\rangle^{\mathrm{mf}}_{d_i,d_j,l_{ij}}}}{\sum_{d_i,d_j,l_{ij}}\mathrm{e}^{-\langle U\rangle^{\mathrm{mf}}_{d_i,d_j,l_{ij}}}} \tag{45}$$

$$\begin{aligned}&\mathcal{B}_d^{\beta_d}(l(i,j),d_i,d_j)\\ &\equiv \sum_{i,j\in\boldsymbol{N}(i)}(1-l(i,j))\parallel d_i-d_j\parallel^2\mathrm{e}^{-\beta_d\|d_i-d_j\|}\end{aligned} \tag{46}$$

$$\begin{aligned}\langle U\rangle^{\mathrm{mf}}_{d_i,d_j,l_{ij}} \equiv{}& B\sum_i(1-\langle s_i\rangle^{\mathrm{mf}})(x_i^{\tau}-x_{i-d_i}^{\tau-1})^2\\ &+ \lambda_d\sum_{i,j\in\boldsymbol{N}(i)}(1-2\,\mathrm{e}^{-\beta_d\|d_i-d_j\|^2})(1-l(i,j))\\ &+ \lambda_s\sum_{i,j\in\boldsymbol{N}(i)}(1-l(i,j))(1-2\delta(\langle s_i\rangle^{\mathrm{mf}}-\langle s_j\rangle^{\mathrm{mf}}))\\ &+ \alpha_l\sum_{i,j\in\boldsymbol{N}(i)}\frac{l(i,j)}{(x_i^{\tau}-x_j^{\tau})^2}+T_s\sum_i\langle s_i\rangle^{\mathrm{mf}}\end{aligned} \tag{47}$$

$$\frac{dT_s}{dt} = -\frac{\sum_{s_i}\{\sum_i s_i\}\mathrm{e}^{-\langle U\rangle^{\mathrm{mf}}_{s_i}}}{\sum_{s_i}\mathrm{e}^{-\langle U\rangle^{\mathrm{mf}}_{s_i}}} \tag{48}$$

$$\begin{aligned}\langle U\rangle^{\mathrm{mf}}_{s_i} \equiv{}& B\sum_i(1-s_i)(x_i^{\tau}-x_{i-\langle d_i\rangle^{\mathrm{mf}}}^{\tau-1})^2\\ &+ \lambda_d\sum_{i,j\in\boldsymbol{N}(i)}(1-2\,\mathrm{e}^{-\beta_d|\langle d_i\rangle^{\mathrm{mf}}-\langle d_j\rangle^{\mathrm{mf}}|^2})\\ &\times (1-\langle l(i,j)\rangle^{\mathrm{mf}})\\ &+ \lambda_s\sum_{i,j\in\boldsymbol{N}(i)}\\ &\times (1-\langle l(i,j)\rangle^{\mathrm{mf}})(1-2\delta(s_i-\langle s_j\rangle^{\mathrm{mf}}))\\ &+ \alpha_l\sum_{i,j\in\boldsymbol{N}(i)}\frac{\langle l(i,j)\rangle^{\mathrm{mf}}}{(x_i^{\tau}-x_j^{\tau})^2}+T_s\sum_i s_i\end{aligned} \tag{49}$$

where $\langle\cdots\rangle^{\rm mf}$ denotes a solution for the corresponding mean-field equation for a given hyper-parameter set at time t of the above learning equations : $\mathbf{\Sigma}^{(t)}$. There still exist several (it is still hard for us to treat by hand) sums in the above learning equations and it might be possible for us evaluate the sums also by the expectations in terms of mean-field approximation. However, for such treatment, the learning equations looks for the hyper-parameters which minimize the cost function instead of the 'negative' marginal likelihood. From the view point of statistical physics, the marginal likelihood corresponds to the negative free energy and the mean-field treatment eliminates the entropy term. Therefore, if we rewrite the marginal likelihood by means of mean-field approximation, one obtains the negative cost function instead of the marginal likelihood. This means that we can not obtain appropriate hyper-parameters in terms of the maximum marginal likelihood criteria. For this reason, here we use the Markov chain Monte Carlo method (MCMC) to evaluate the sums appearing in the right hand sides of the learning equations.

In order to implement the learning equations in computer, we discretize the derivative with respect to time t by means of Euler method such as

$$B(t+\Delta t) = B(t) + \Delta t \left\{ \frac{\sum_{d_i,s_i}\left\{\sum_i (1-s_i)(x_i^{\tau}-x_{i-d_i}^{\tau-1})^2\right\} e^{-\langle U\rangle_{s_i,d_i}^{\rm mf}}}{\sum_{d_i,s_i} e^{-\langle U\rangle_{s_i,d_i}^{\rm mf}}}\right\}.$$

Thus, we set the initial values of hyper-parameters to $\mathbf{\Sigma}^{(0)}$ and solve the mean-field equations. Then, we insert the solutions into the right hand sides of the above learning equations and evaluate the sums such as $\sum_{s_i}(\cdots)$ by the MCMC. After that, we update the hyper-parameters by the discretized learning equations and also update the time (step) as $t \mapsto t+1$. We repeat these procedures until each hyper-parameter converges to some finite value. Here we set $\Delta t = 0.001$. The initial values $\mathbf{\Sigma}^{(0)}$ are the same values as those by Zhang and Hanouer (1995).

In Fig. 4, we show the typical snapshots of velocity fields obtained by the method of hybridization of mean-field approximation and MCMC at time $t=0$ (upper left)C $t=10$ (upper right), $t=20$ (lower left), $t=30$ (lower right)C The case of $t=0$ corresponds to the result by Zhang and Hanouer (1995). From these panels, we find that our approach remarkably improves the performance of Zhang and Hanouer (1995).

1) Average-case performance measures: To evaluate the average-case performance more quantitatively, we introduce the following two kinds of performance measures. The first one is defined by

$$K \equiv \frac{1}{N}\sum_i (1-\cos\theta_i) \tag{50}$$

where θ_i denotes an angle between the true velocity vector fields $\boldsymbol{d}^0 = \{\vec{d}_1^0,\cdots,\vec{d}_N^0\}$ and the estimated fields $\boldsymbol{d} = \{\vec{d}_1,\cdots,\vec{d}_N\}$, that is explicitly given by $\cos\theta_i = \vec{d}_i^0\cdot\vec{d}_i/\parallel \vec{d}_i^0 \parallel\parallel \vec{d}_i \parallel$. From the above definition, the K measures the error concerning mismatch of the direction of the estimated vector.

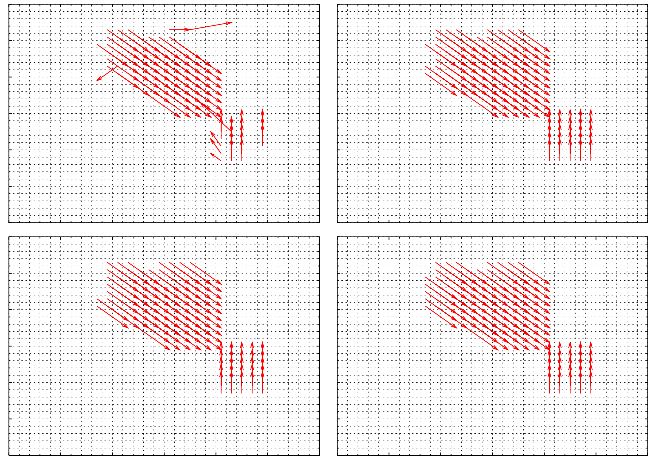

Fig. 4. Typical snapshots of velocity fields obtained by the method of hybridization of mean-field approximation and MCMC at time $t=0$ (upper left)C $t=10$ (upper right), $t=20$ (lower left), $t=30$ (lower right)C The case of $t=0$ corresponds to the result by Zhang and Hanouer (1995).

Besides of the above K, we next introduce

$$L \equiv \frac{1}{N}\sum_i \left(1-\frac{\parallel \vec{d}_i \parallel}{\parallel \vec{d}_i^0 \parallel}\right) \tag{51}$$

which measures the error concerning mismatch of the length of the estimated vector. We show the results in Fig. 5. We plot the average values of K and L over 20-independent runs for various different choices of the successive two video-frames. From these two panels, we find that these two errors decreases monotonically on average during the proposed learning procedures.

2) Computational cost measure: We next evaluate the computational cost. Obviously, our procedure requires us to take much longer time in comparison with the result by Zhang and Hanouer (1995) to obtain the results because for each Euler step, one needs to solve the mean-field equations and one should carry out the MCMC at the same time. In Fig. 6, we plot the CPU time CT [sec] as a function of system size N. The CPU time is measured in our PC (*DELL Optiplex960DT7, Core2QuadQ9400 2.66 GHz*). In the case of Zhang and Hanouer (1995), we measure the CT [sec] as CPU time to proceed 50-times mean-field iterations, whereas, in the case of our proposed procedure, the CT is defined as CPU time to take $t=50$ in learning equations (for each of t, 50-times mean-field iterations and 100 Monte Carlo step are done). From Fig. 6, we find that the difference between two procedures increases exponentially, however, this fact does not mean that our proposed procedure is computationally inferior to the ad-hoc choice by Zhang and Hanouer (1995) because they found the value by 'try and error' manner and it might take a quite long time to determine the value although they did not mention this point explicitly in their paper.

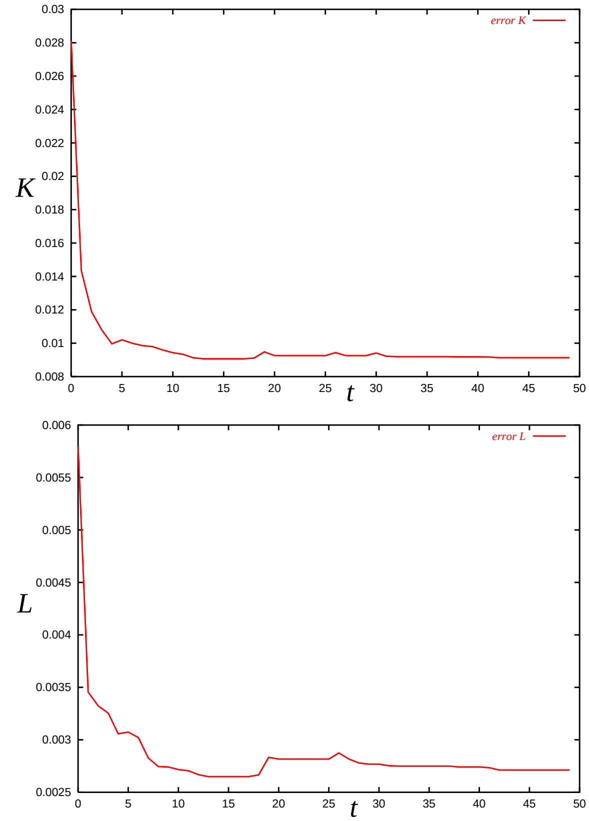

Fig. 5. Time dependence of the performance measures K (upper panel) and L (lower panel). We plot the average values of K and L over 20-independent runs for various different choices of the successive two video-frames.

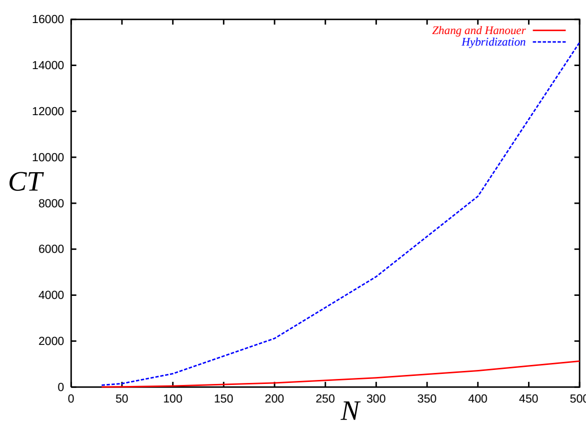

Fig. 6. Computational time (real CPU time) CT [sec] until the algorithm converges as a function of system size N.

C. Simple MCMC approach

In general, the preciseness of the mean-field approximation is not so good. Here we attempt to use simple MCMC instead of hybridization of mean-field approximation and MCMC to calculate the expectations of quantities appearing in the learning equations over the posterior. Then, we compare the results with those obtained by hybridization of the mean-field approximation and the MCMC discussed in the previous subsection.

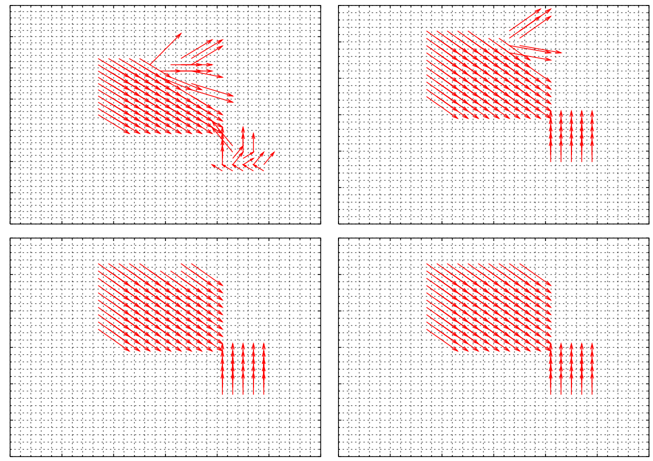

Fig. 7. Typical snapshots of velocity fields obtained by the method of simple MCMC at time $t = 0$ (upper left)C $t = 10$ (upper right), $t = 20$ (lower left), $t = 30$ (lower right)C The case of $t = 0$ corresponds to the result by Zhang and Hanouer (1995).

We show the results in Fig. 7. From these panels, we find that the resultant velocity fields at $t = 30$ are much closer to the true fields than the result obtained by the hybridization.

We also evaluate the performance measures K, L and compare the results with the results by the hybridization of mean-field approximation and MCMC in Fig. 8. From these two panels, we find that at the initial stage of the learning steps, the hybridization decreases the two kinds of errors very quickly, however, eventually the errors are saturated. On the other hand, the errors by the simple MCMC does not decreases so quickly at the initial stage, however, the resultant errors converge to lower values than those of the hybridization.

We also compare the computational time until the convergence for hybridization and simple MCMC. The result is shown in Fig. 9. From this figure, we notice that the hybridization takes much longer time to proceed than the simple MCMC does because the Monte Carlo steps in the MCMC for each learning step t are the same as the hybridization.

Finally we list the table to compare the hyper-parameters obtained by our methods and by Zhang and Hanouer (1995). We show the result in TABLE I. This table tells us that

	Zhang and Hanouer (1995)	Hybridization	simple MCMC
λ_s	2	2.3	2.5
B	5	12.1	11.7
λ_d	2.5	2.7	2.8
β_d	4	3.8	3.7
α_l	200	232	220
T_s	5	5	5

TABLE I
COMPARISON OF THE RESULTANT HYPER-PARAMETERS.

several parameters in Zhang and Hanouer (1995) are very close to ours or exactly the same as ours, however, some of the parameters are quite far from our results. This means that

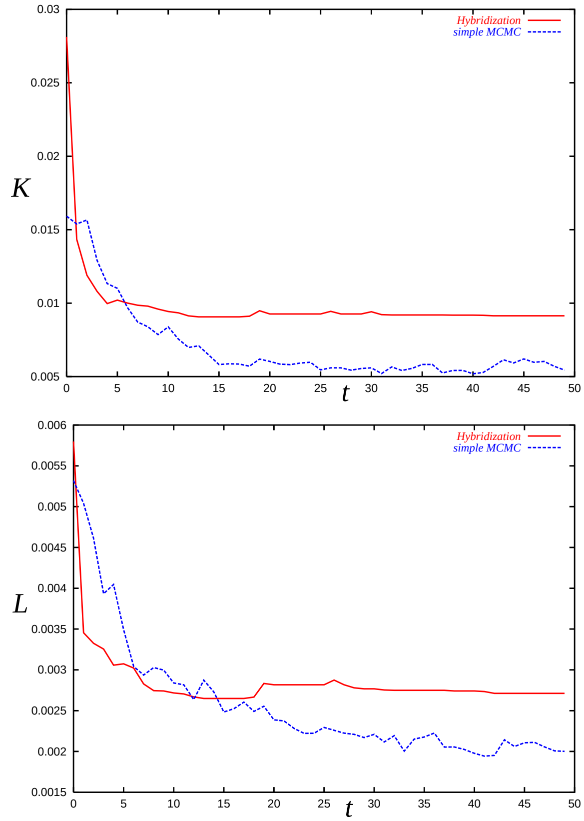

Fig. 8. The Euler step dependence of K (upper panel) and L (lower panel) for the hybridization (solid line) and simple MCMC (broken line).

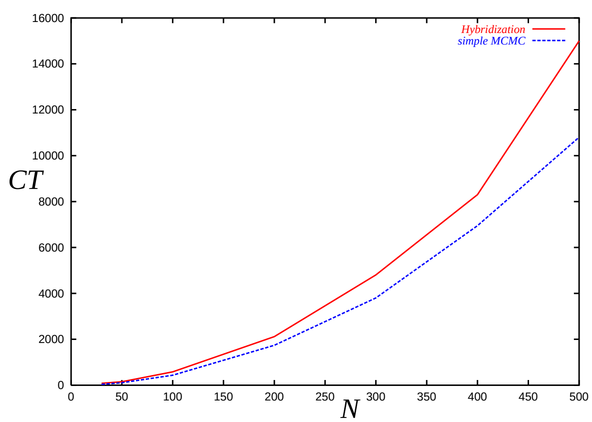

Fig. 9. Computational time (real CPU time) CT [sec] for the hybridization (solid line) and simple MCMC (broken line) as a function of system size N.

the ad-hoc choice by Zhang and Hanouer (1995) is statistically (theoretically) incorrect and if one needs to choose statistically 'proper' hyper-parameters 'systematically', he (or she) should utilize the procedures provided by us in this paper.

VII. Summary

In this paper, we numerically examined a Bayesian mean-field approach with the assistance of the MCMC method to estimate motion velocity fields and probabilistic models simultaneously in consecutive digital images described by spatio-temporal Markov random fields. We found that our motion estimation is much better than the result obtained by Zhang and Hanouer (1995) in which the hyper-parameters are set to some ad-hoc values without any theoretical justification.

Utilization of EM algorithm to determine the hyper-parameters by maximizing the marginal likelihood indirectly [12], [13], analytical evaluation of the average-case performance by making use of mathematically solvable MRFs such as Gaussian MRFs [14] or infinite range MRFs [12], applying the Belief propagation [15] to compute the marginal probability in our framework are now on going and the results will be reported in the conference or elsewhere.

Acknowledgment

We were financially supported by Grant-in-Aid Scientific Research on Priority Areas *'Deepening and Expansion of Statistical Mechanical Informatics (DEX-SMI)'* of the MEXT No. 18079001. One of the authors (JI) was financially supported by INSA (Indian National Science Academy) - JSPS (Japan Society of Promotion of Science) Bilateral Exchange Programme. He also thanks Saha Institute of Nuclear Physics for their warm hospitality during his stay in India. We acknowledge Professor Tsuyoshi Horiguchi for drawing our attention to the reference [7] when he visited our research group in 2004.

References

[1] S. Geman and D. Geman, *IEEE Transaction on Patt. Anal. Machine Intell.*, **PAMI-6**, pp. 721-741 (1984).
[2] A.N. Netravali, *Digital Pictures: Representation and Compression*, New York: Plenum (1988).
[3] G. Winkler, *Image Analysis, Random Fields and Markov Chain Monte Carlo Methods*, Springer (2003).
[4] C.M. Bishop, *Pattern Recognition and Machine Learning*, Springer (2006).
[5] S.C. Park, M.K. Park and M.G. Kang, *IEEE Signal Process. Mag.*, **20**, N0.3, pp. 21-36 (2003).
[6] A. Kanemura, S. Maeda and S. Ishii, *Proc. ICANN, LNCS*, **4669**, pp. II-611-620, Springer (2007).
[7] J. Zhang and G.G. Hanauer, *IEEE Transaction on Image Processing*, **40**, No.1, pp. 19-33 (1995).
[8] J. Wei and Z.-N. Li, *IEEE Transaction on Circuits and Systems for Video Technology*, **9**, No. 6, pp. 960-972 (1999).
[9] A. Caplier, F. Luthon and C. Dumontier, *Real-Time Imaging*, **4**, pp. 41-54 (1998).
[10] F. Luthon, A. Caplier and M. Lievin, *Signal Processing*, **76**, pp. 61-80 (1999).
[11] M. Mezard, G. Parisi and M.A. Virasoro, *Spin Glass Theory and Beyond*, World Scientific, Singapore (1987).
[12] J. Inoue and K. Tanaka, *Physical Review E*, **65**, 016125 (2002).
[13] K. Tabushi and J. Inoue, *Proceedings of the IEEE Neural Networks for Signal Processing*, **XI**, pp.133-142 (2001), J. Inoue and K. Tabushi, *International Journal of Modern Physics B*, **17**, No. 29, pp. 5525-5539 (2003).
[14] K. Tanaka and J. Inoue, *IEICE Trans. on Information and Systems* **E85-D**, No. 3, pp. 546-557 (2002).
[15] K. Tanaka, J. Inoue and D. M. Titterington, *Journal of Physics A : Mathematical and General*, **36** pp. 11023-11035 (2003).